



**FACULTAD DE INGENIERÍAS**

## Monitoreo de una cámara térmica a través de un aplicativo móvil

Trabajo de grado presentado por:

**Lina Marcela Rodríguez Soto**

Para optar por el título de:

**Ingeniera Electrónica**

Director del trabajo de grado:

**M.Sc. Arley Bejarano Martínez**

# Monitoreo de una cámara térmica a través de un aplicativo móvil

LINA MARCELA RODRÍGUEZ SOTO

UNIVERSIDAD TECNOLÓGICA DE PEREIRA

Facultad de ingenierías

Programa de Ingeniería electrónica

Pereira, marzo 2019

## ***Dedicatoria***

*Dedico este trabajo a mí adorada hija Caroline Manuela, por ser mi motivación y mi gran orgullo, porque ha sido mi inspiración y a pesar de su corta edad su apoyo y compañía fue determinante para culminar este proceso.*

## **AGRADECIMIENTOS**

Doy gracias a Dios por permitirme culminar este proceso, a mí amado esposo por su incondicional apoyo, por ser mi fuente de inspiración cada día. A mis padres por su amor, a mi hermana por sus consejos y apoyo en todo momento.

Un especial agradecimiento a mi director de tesis M.Sc. Arley Bejarano Martínez quien me orientó durante el desarrollo de este trabajo, gracias por su paciencia, por su profesionalismo, pero sobre todo por la pasión y entrega que tiene con los estudiantes y el sentido de pertenencia con el programa de ingeniería electrónica.

Gracias a la Universidad Tecnológica de Pereira, por permitirme ser parte de ella, por darme la oportunidad de formarme profesionalmente, doy gracias a los docentes que hicieron parte de este proceso, por la formación y los conocimientos brindados, porque hicieron posible este logro tan importante en mi vida.

Gracias a todos mis compañeros, en especial a mi amigo Juan Sebastián Carmona, porque fue un apoyo en los momentos más difíciles de mi vida, tanto personales como académicos. También agradezco a mi compañero Sheiner Alberto Sánchez, por sus enseñanzas, paciencia y apoyo a lo largo de la carrera.

Por último gracias a la facultad de ciencias agrarias y agroindustria de la UTP, por facilitarme el espacio en el corregimiento La bella para realizar las pruebas del trabajo de grado.

## Tabla de Contenido

Índice de figuras	6
Resumen	7
Capítulo 1. Preliminares	8
1.1 Definición del problema	8
1.2 Objetivos	9
1.2.1 Objetivo general	9
1.2.2 Objetivos específicos	9
Capítulo 2. Marco teórico.	10
2.1 Temperatura	10
2.2 Humedad Relativa	10
2.3 Sensor DHT22	10
2.4 BeagleBone Black	11
2.5 Android Studio	12
2.6 Kotlin	13
2.7 XAMPP	13
Capítulo 3. Estado del arte	14
Capítulo 4. Desarrollo	15
4.1 Hardware	15
4.1.1 Sistema de adquisición	15
4.1.2 Sistema de telemetría	16
4.1.3 Sistema de la información	16
4.1.4 Interconexión sistema de telemetría y sistema de información	17
4.1.5 Diagrama de flujo	18
4.2 Software	18
4.2.1 Descripción del aplicativo móvil	20
4.2.2 Interfaz gráfica y descripción de botones	20
Capítulo 5. Resultados	25
Capítulo 6. Conclusiones	29
Capítulo 7. Trabajos futuros	30
Bibliografía	31
Anexos	33

## Índice de figuras

<u>Figura 1. Sensor digital DHT22 de temperatura y humedad relativa. Tomado de <a href="https://ugedone.com/dht22-temphumidity-sensor-module.html">https://ugedone.com/dht22-temphumidity-sensor-module.html</a> .....</u>	<u>11</u>
Figura 2. BeagleBone Black.....	12
Figura 3. Diagrama funcional del desarrollo de hardware .....	15
Figura 4. Conexión sensor DHT22.....	15
Figura 5. Diagrama metodológico primera etapa .....	16
Figura 6. Diagrama metodológico segunda etapa .....	16
Figura 7. Estructura de la información de la base de datos con información del cultivo. ....	17
Figura 8. Panel de control de XAMPP .....	17
Figura 9. Diagrama de funcionamiento sistema de telemetría y el de la información. ....	18
Figura 10. Diagrama metodológico tercera etapa.....	19
Figura 11. Diagrama funcional del desarrollo de software .....	19
Figura 12. Interfaz gráfica principal del aplicativo móvil .....	20
Figura 13. Segunda interfaz gráfica principal del aplicativo móvil .....	22
Figura 14. Resultado de búsqueda en el aplicativo móvil. ....	23
Figura 15. Diagrama del circuito implementado. ....	24
Figura 16. Implementación física en el invernadero Horfrubella.....	25
Figura 17. Base de datos con los valores obtenidos del cultivo. ....	26
Figura 18. Resultado de búsqueda para la variable temperatura. ....	26
Figura 19. Resultado de búsqueda para la variable humedad relativa .....	27
Figura 20. Resultado de búsqueda cuando no hay conexión. ....	27
Figura 21. Resultado de búsqueda cuando no hay datos .....	28

## **RESUMEN**

La tendencia en los últimos años de mejorar la producción agrícola en nuestro país, ha permitido desarrollar técnicas con el fin de cuidar los cultivos de enfermedades, plagas y de consecuencias negativas generadas por las variaciones climáticas. Los campesinos han optado por proteger los cultivos a través de cámaras térmicas, las cuales traen beneficios en la agricultura, ya que favorecen la proliferación de plántulas para incrementar la producción, permiten establecer condiciones climáticas óptimas, uso eficiente de los insumos y la posibilidad de cultivar durante todo el año. Si bien sabemos, los costos de la implementación de tecnologías agrícolas y de insumos son muy altos, por lo tanto cada vez son más los campesinos que abandonan esta actividad. Lo que se pretende en el desarrollo de este trabajo, es ofrecer una herramienta tecnológica, que traiga beneficios a la región y que sea a un bajo costo.

Tomando como punto de partida de que en el territorio nacional, la mayoría de personas cuentan con un dispositivo móvil y conectividad a internet gracias a las zonas wifi implementadas por el gobierno, se propone desarrollar un aplicativo móvil, que permite monitorear las variables físicas de temperatura y humedad relativa en las cámaras térmicas, la finalidad es que los campesinos puedan acceder a ella, de tal forma que la interfaz gráfica sea fácil de utilizar.

El desarrollo se divide en dos partes, hardware y software, en la primera parte requiere para su implementación, un sistema de telemetría, un sistema de la información y finalmente el diseño y creación del aplicativo móvil.

## **1. PRELIMINARES**

### **1.1 Definición del problema.**

La agricultura es una de las actividades más antiguas del sector primario, se estima su origen en la edad de piedra (año 7000 A.C), donde las sociedades humanas pasaron del nomadismo al sedentarismo, formando el inicio de las civilizaciones [1]. En América, los pueblos Aztecas y los mayas iniciaron la explotación agrícola sembrando maíz, yuca, cacao, piña y frijoles, mientras que los Incas cultivaban papas [2]. Hoy en día sigue siendo una actividad importante en la economía de nuestro país, ya que incentiva el desarrollo social y gestión sostenible de los recursos naturales.

Colombia es un país con excelente ubicación geográfica y cuenta con grandes riquezas naturales [3], a pesar de ello, cada vez son más los campesinos que dejan las actividades del campo, ya que no cuentan con las condiciones para producir más y mejor, debido a los altos costos de insumos y servicios. De acuerdo con César Ferrari, economista de la Universidad Javeriana, en la actualidad de los 41.5 millones de hectáreas del territorio nacional destinadas al sector agropecuario, tan solo 7.1 millones de hectáreas están pensadas para cultivos, mientras que los restantes 34.4 millones se destinan a la ganadería extensiva, que en comparación con los cultivos agrícolas muy pocos campesinos se benefician [4].

Los cultivos se ven afectados por las variaciones climáticas, que cada vez son más adversas [5], por ello la tecnología juega un papel importante para generar una producción más eficiente y con un menor impacto ambiental. Existen diferentes tecnologías en la agricultura que ayudan a mejorar la producción, como lo es el monitoreo de cultivos a través de la termografía, la cual se basa en la detección de radiación infrarroja emitida por los cuerpos, convirtiendo la información obtenida en imágenes visibles con información sobre la temperatura de la superficie de los mismos[6]; también ha tenido gran relevancia el uso de drones, ya que por medio de las imágenes captadas se pueden realizar diferentes actividades que ayudan al diagnóstico de adversidades, tales como detección de enfermedades, control de temperaturas y sistema de riego, esta tecnología genera grandes aportes en el desarrollo agrícola [7], pero implica altos costos y los campesinos colombianos no tienen los recursos suficientes para adquirirlas.

A pesar de que existen diferentes Tratados de Libre Comercio TLC, la agricultura colombiana ha tenido un retraso significativo en innovaciones tecnológicas, ya que los campesinos no cuentan con los recursos, la tecnología y capacitación suficientes para ser competitivos en el mercado global.

¿Es posible desarrollar un aplicativo móvil que permita monitorear una cámara térmica de cultivos?



Esto traería beneficios para el desarrollo agrícola de nuestro país, ya que por medio de la creación de un aplicativo móvil, los agricultores pueden llevar en su celular la información de los cultivos, teniendo en cuenta que en el territorio nacional la mayoría de las personas cuentan con dispositivos móviles y conectividad a internet, gracias al ministerio de tecnologías de la información y la comunicación TIC, que han entregado más de 1400 zonas Wifi en todo Colombia [8]. Al aplicar tecnología en los cultivos, hace que el país se vuelva competitivo a nivel global, ya que los productos cumplirían con estándares de calidad y de este modo se aprovecharían los tratados de libre comercio, al producir más y mejor se podría exportar generando mejoras en la economía en este importante sector primario. Con lo anterior se puede mejorar la calidad de vida de los agricultores y de esta manera que ellos vuelvan a las actividades de campo.

## **1.1 Objetivos**

### **1.1.1 Objetivo general**

Diseñar un sistema de monitoreo de variables a través de un aplicativo móvil.

### **1.1.2 Objetivos específicos**

- Elaborar un sistema de telemetría para obtener las variables físicas de humedad relativa y de temperatura.
- Crear un sistema de almacenamiento de los datos obtenidos.
- Diseñar un aplicativo móvil que permita el monitoreo de los datos.

## **2. MARCO TEÓRICO**

### **2.1 Temperatura**

La temperatura es una magnitud física, que refleja la cantidad de calor que tiene un cuerpo, objeto o ambiente, las partículas que poseen los cuerpos se mueven a una determinada velocidad, por lo que cada una cuenta con una determinada energía cinética. El valor medio de dicha energía cinética  $\langle E_c \rangle$  está directamente relacionado con la temperatura del cuerpo. Así, a mayor energía cinética media de las partículas, mayor temperatura y a menor energía cinética media, menor temperatura [9].

La temperatura es esencial en los cultivos, ya que influye en el crecimiento y productividad de las plantas, la temperatura de la planta y la del ambiente no son iguales porque las plantas son capaces de enfriarse por evaporación y de calentarse por irradiación. Las plantas buscan alcanzar su temperatura óptima, para lo que es muy importante que exista un equilibrio entre la temperatura ambiental, la humedad relativa y la luz [10].

### **2.2 Humedad relativa**

El vapor de agua permite regular la temperatura del ambiente, absorbiendo la radiación térmica del sol y de la tierra, por lo tanto la humedad relativa es la relación entre la cantidad de vapor de agua que existe en un ambiente con la cantidad de vapor de agua que podría estar contenida en su máximo punto en ese aire a una temperatura determinada.

La humedad es uno de los factores que más influye en las cámaras térmicas, ya que en su interior, el aire es enriquecido con vapor de agua, obtenido por evaporación desde el suelo y por la transpiración de los cultivos, la transpiración se realiza ya que de esa manera es que las plantas transportan los nutrientes y se enfrían para su crecimiento y su respectiva regulación [11].

### **2.3 Sensor DHT22**

El sensor DHT22 de salida digital, tiene incorporada la tecnología de detección para garantizar la fiabilidad y estabilidad, recoge señales digitales de humedad relativa y temperatura, y las calibra mediante una cámara de calibración precisa, el coeficiente obtenido, lo almacena en la memoria OTP, cuando el sensor está detectando, cita dicho coeficiente [12]. Utiliza un sensor capacitivo de humedad y un termistor para medir el aire.

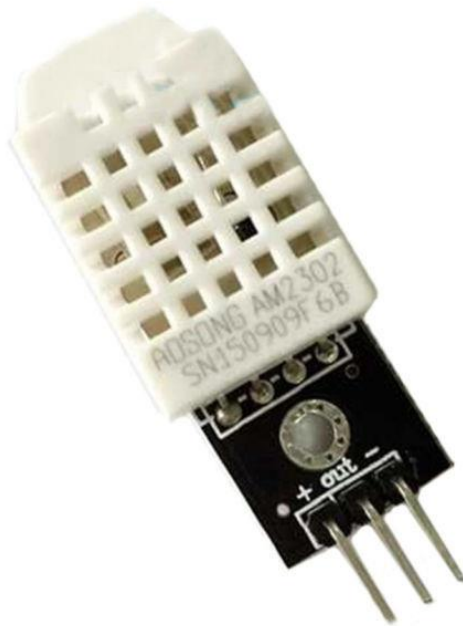


Figura 1. Sensor digital DHT22 de temperatura y humedad relativa. Tomado de <https://uged-one.com/dht22-temphumidity-sensor-module.html>

#### **Características:**

- Fuente de alimentación: 3.3-6V DC.
- Rango de medición de temperatura: -40°C a 80 °C.
- Precisión de medición de temperatura:  $\pm 0.5^{\circ}\text{C}$ .
- Precisión de medición de humedad:  $\pm 2\% \text{ RH}$  (Max  $\pm 5\% \text{ RH}$ )
- Resolución o sensibilidad de humedad: 0.1% HR.
- Resolución o sensibilidad de temperatura: 0.1°C
- Rango de medición de humedad: De 0 a 100% RH
- Distancia de transmisión: 20m.

## **2.4 BeagleBone Black**

Es un sistema embebido de bajo costo y bajo consumo eléctrico que cuenta con un sistema operativo llamado Angstrom Linux [13].

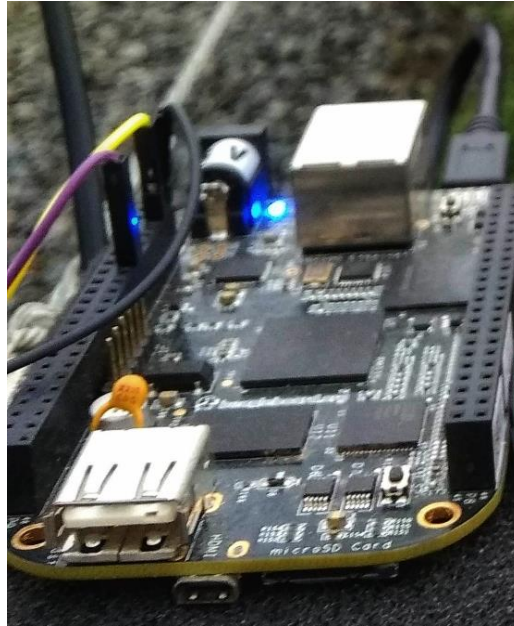


Figura 2. BeagleBone Black.

### **Características**

- Procesador AM335x ARM A8 a 1GHz.
- 512MB de memoria RAM DDR3.
- Gráfica SGX530 para procesar gráficos en 3D.
- Salida de audio y vídeo a través del puerto mini-HDMI.
- Tarjeta de red Fast Ethernet 10/100Mbps.
- 1 puerto USB 2.0.
- 2GB de memoria interna (para el Sistema Operativo) y la posibilidad de ampliarla mediante una memoria USB o una micro-SD.
- Dispone de botones de encendido, reset y boot.

## **2.5 Android Studio**

Android Studio es el entorno de desarrollo integrado (IDE) oficial para el desarrollo de aplicaciones para Android y se basa en IntelliJ IDEA. Además del potente editor de códigos y las herramientas para desarrolladores de IntelliJ, Android Studio ofrece aún más funciones que aumentan la productividad durante la compilación de apps para Android [14]

### **Características**

- Integración de ProGuard y funciones de firma de aplicaciones.

- Renderizado en tiempo real
- Consola de desarrollador: consejos de optimización, ayuda para la traducción, estadísticas de uso.
- Soporte para construcción basada en Gradle.
- Refactorización específica de Android y arreglos rápidos.
- Un editor de diseño enriquecido que permite a los usuarios arrastrar y soltar componentes de la interfaz de usuario.
- Herramientas Lint para detectar problemas de rendimiento, usabilidad, compatibilidad de versiones y otros problemas.
- Plantillas para crear diseños comunes de Android y otros componentes.
- Soporte para programar aplicaciones para Android Wear.
- Soporte integrado para Google Cloud Platform, que permite la integración con Google Cloud Messaging y App Engine.
- Un dispositivo virtual de Android que se utiliza para ejecutar y probar aplicaciones.

## **2.6 Kotlin**

Es un lenguaje de programación tipo estático desarrollado por la empresa JetBrains en sus oficinas de San Petersburgo, Rusia. A pesar de ser un lenguaje nuevo, ha sido aceptado oficialmente por Google para la plataforma de Android Studio. Presenta ventajas en comparación con Java, aunque se puede operar utilizando los módulos desarrollados en Java, y con otros módulos desarrollados en Kotlin, ya que estos dos pueden comunicarse entre sí sin presentar inconvenientes. Una de las ventajas es que Kotlin permite expresar las ideas de programación en menos líneas, lo cual implica menos trabajo a la hora de probarlas y mantenerlas [15]. Cuando se realiza la compilación con Kotlin, se genera un código JVM (Java Virtual Machine) el cual pasa a ser interpretado por una máquina virtual de Java, gracias a su compatibilidad. Otra ventaja que presenta Kotlin respecto a Java, es que al ser un lenguaje nuevo, presenta características que le permiten al programa ser más seguro y conciso.

## **2.7 XAMPP**

Es un servidor de plataforma libre que integra en una sola aplicación, un servidor de web Apache, intérpretes de lenguajes script PHP, un servidor de base de datos MySQL, entre otros.

Una de las características principales de XAMPP, es que es multiplataforma, es decir, existen versiones para diferentes sistemas operativos, tales como: Microsoft Windows, GNU/Linux, Solaris, y MacOS X. Otra ventaja es que trae PhpMyAdmin para administrar las bases de datos de MySQL.[16]

### 3. ESTADO DEL ARTE

Actualmente existen diferentes tecnologías que se han implementado para el monitoreo de cultivos, esto es debido al crecimiento y la participación en el área de agricultura de precisión. Un ejemplo de ellos es el monitoreo de cultivos a través de la termografía, la cual se basa en la detección de radiación infrarroja emitida por los cuerpos, un trabajo donde se utilizó este método fue realizado por una estudiante de la Universidad Nacional en Bogotá, llamado Implementación de imágenes termográficas para la detección de estrés hídrico en hierbabuena (*Mentha spicata*) bajo invernadero en la Sabana de Bogotá [] en este trabajo para realizar el monitoreo se utilizó la cámara termográfica Flir Systems ® Modelo A40M, a pesar de su excelente funcionamiento, su costo es elevado y esto limita a que los agricultores colombianos puedan acceder a ella.

Otro sistema de monitoreo es mediante la utilización de drones, este estudio lo han llevado a cabo investigadores del Grupo de Robótica y Cibernética (RobCib) y del Centro de Automática y Robótica (CAR), que es un centro mixto de la Universidad Politécnica de Madrid (UPM) y del Consejo Superior de Investigaciones Científicas (CSIC), en el cual se puede ayudar a mejorar los sistemas de control del clima y la monitorización de los cultivos, ya que la capacidad del robot aéreo para desplazarse en las tres dimensiones y colocar los sensores en cualquier punto supone una ventaja sobre otras alternativas como las redes de sensores [17].

En la Universidad Tecnológica de Pereira, se han adelantado trabajos en el área de la agricultura de precisión, uno de ellos corresponde al diseño e implementación de un prototipo de una red de sensores inalámbricos para cultivos de frutas utilizando Beaglebone, el cual permite almacenar los datos del estado del cultivo en una base de datos. A pesar de ser un muy buen trabajo desarrollado, no cuenta con una interfaz de usuario que permita visualizar la información. Otro trabajo de grado, permite el monitoreo y control de un cultivo en invernadero, presenta además otras variables del cultivo, como humedad del suelo y luminosidad, permite descargar la información de la base de datos en Excel y visualizar los gráficos a través de una página web, si bien es un avance significativo, carece de portabilidad para que los usuarios tengan acceso de manera inmediata, sin tener que esperar conectarse a un computador y realizar la consulta.[20]

## 4. DESARROLLO

### 4.1 Hardware



Figura 3. Diagrama funcional del desarrollo de hardware

#### 4.1.1 Sistema de adquisición

En la figura 4, se puede observar el sensor DHT22 con sus cables de conexión, el cable de color negro va conectado a la tierra, el cable rojo a la alimentación de 3-5V y el de color verde, va conectado al pin de entrada de datos del sistema.

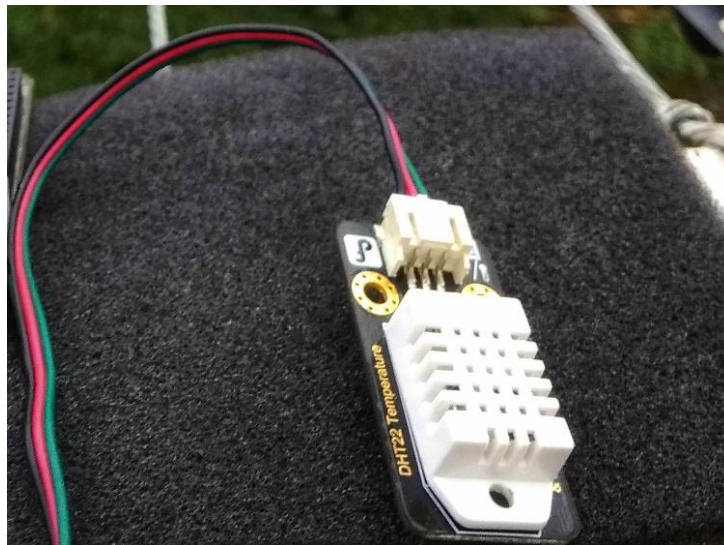


Figura 4. Conexión sensor DHT22

#### 4.1.2 Sistema de telemetría

Para obtener las mediciones remotas de las magnitudes físicas de temperatura y humedad relativa, se procede a realizar un código en C++ para que el módulo BeagleBone Black, se pueda con el sensor DHT22 y de esta manera hacer las mediciones; para ello fue preciso

utilizar las librerías de Adafruit Industries[]. Luego dichas medidas, se envían al servidor. [ver anexo código.cpp]

Para el diseño del sistema de telemetría con el fin de obtener las variables físicas de humedad relativa temperatura, se llevaron a cabo las siguientes actividades descritas que se pueden observar a continuación en el diagrama:

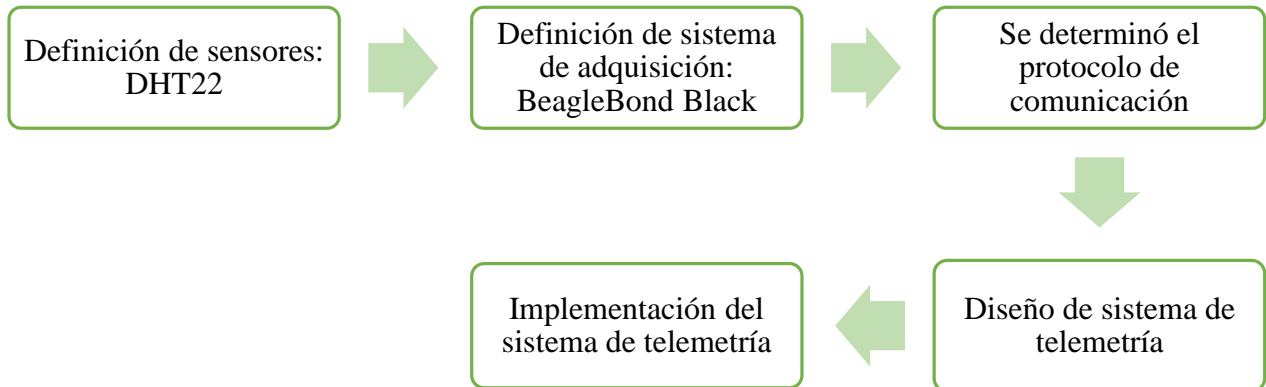


Figura 5. Diagrama metodológico primera etapa.

#### 4.1.3 Sistema de la información

Con el fin de manejar la administración y la gestión de la base de datos de los valores obtenidos, se utiliza la herramienta phpMyAdmin escrita en PHP, y se hace la creación de la base de datos.

En la figura 6, se aprecia el desarrollo metodológico para almacenar los datos adquiridos:

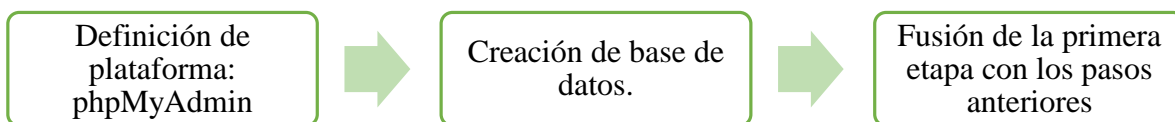


Figura 6. Diagrama metodológico segunda etapa.

A continuación en la figura 7, se puede observar la estructura de la información del cultivo tales como, id que corresponde al número del registro, temperatura y humedad que son las variables que se desean conocer del cultivo y la fecha de registro del dato.



Estructura de tabla Vista de relaciones

#	Nombre	Tipo	Cotejamiento	Atributos	Nulo	Predeterminado	Comentarios	Extra	Acción
<input type="checkbox"/> 1	id	int(10)			No	Ninguna		AUTO_INCREMENT	Cambiar  Eliminar  Más
<input type="checkbox"/> 2	temperatura	varchar(5)	latin1_spanish_ci		No	Ninguna			Cambiar  Eliminar  Más
<input type="checkbox"/> 3	humedad	varchar(5)	latin1_spanish_ci		No	Ninguna			Cambiar  Eliminar  Más
<input type="checkbox"/> 4	fecha	timestamp			No	CURRENT_TIMESTAMP			Cambiar  Eliminar  Más

☐ Seleccionar todo Para los elementos que están marcados: Examinar Cambiar Eliminar Primaria Único Índice  
 Texto completo Agregar a columnas centrales Eliminar de las columnas centrales

Figura 7. Estructura de la base de datos con la información del cultivo.

#### 4.1.4 Interconexión del sistema de telemetría con el de la información

Para almacenar el dato en la base de datos se utilizó la plataforma XAMPP.

En la figura 8, se observa que para dar inicio al servidor, se debe seleccionar Start en el módulo Apache y en MySQL.

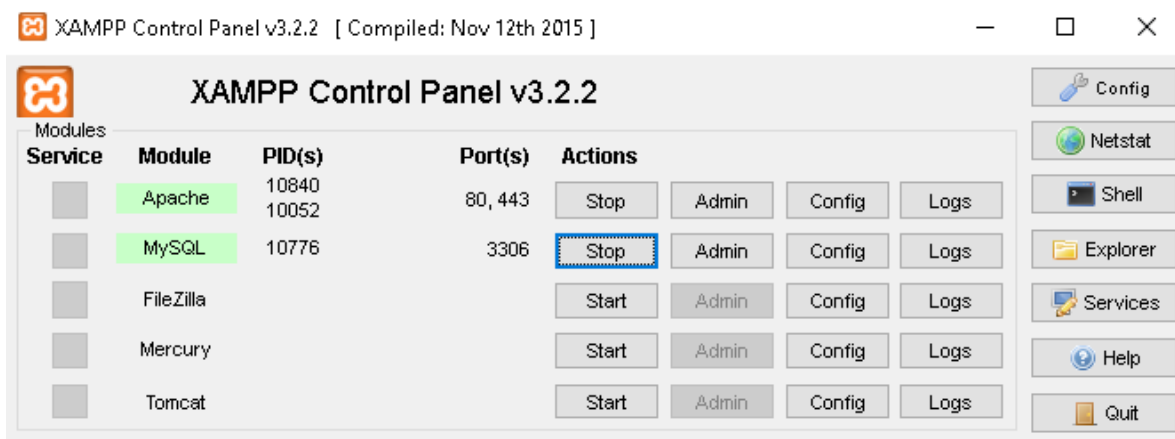


Figura 8. Panel de control de XAMPP.

#### 4.1.5 Diagrama de flujo

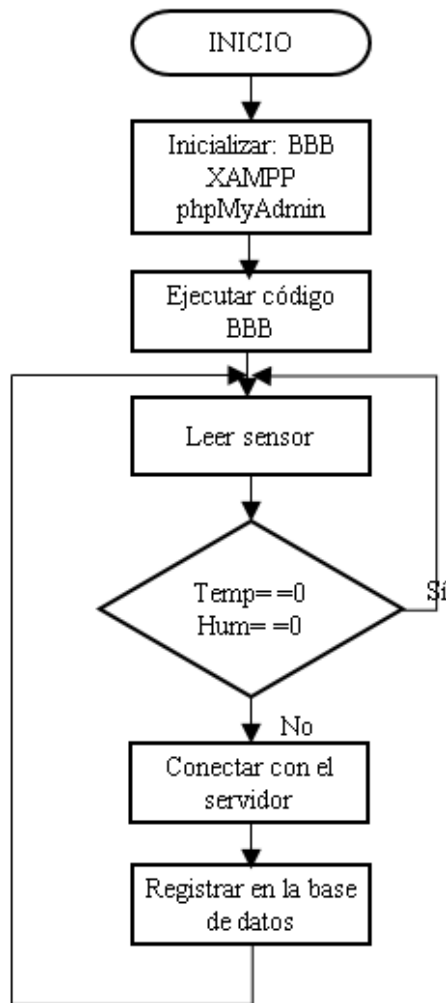


Figura 9. Diagrama de funcionamiento sistema de telemetría y sistema de la información.

#### 4.2 Software

La elección del sensor se realizó teniendo en cuenta su bajo costo y la precisión para obtener las medidas de temperatura y humedad relativa, partiendo de lo anterior, se encontró que los sensores digitales tienen mayor fidelidad, es decir, que poseen la capacidad de dar el mismo valor de la medida, al medir varias veces en unas mismas condiciones determinadas, los sensores digitales también presentan muchas veces mayor exactitud, esto es que sus medidas se aproximan al valor verdadero de la magnitud medida.[libro]

Tomando los criterios anteriormente mencionados descritos en el capítulo 2, se eligió el sensor DHT22

Finalmente en la etapa 3, se realizan las siguientes labores para realizar el monitoreo del cultivo a través del aplicativo móvil:

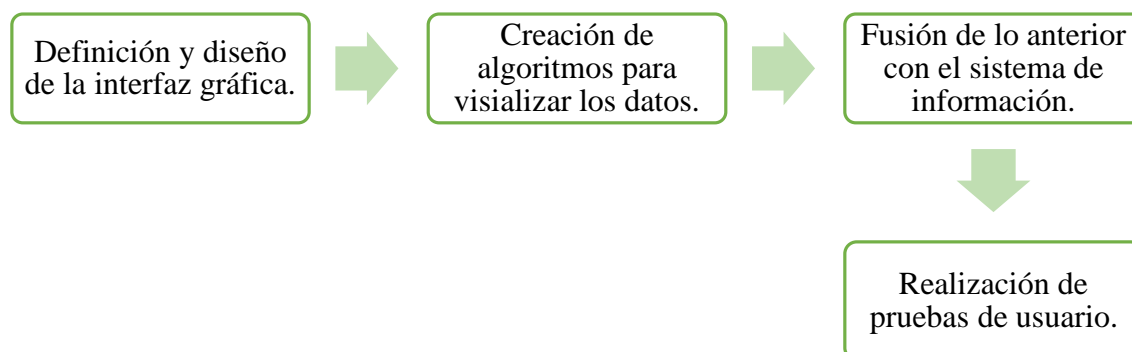


Figura 10. Diagrama metodológico de la tercera etapa.

## 4.2 Software.

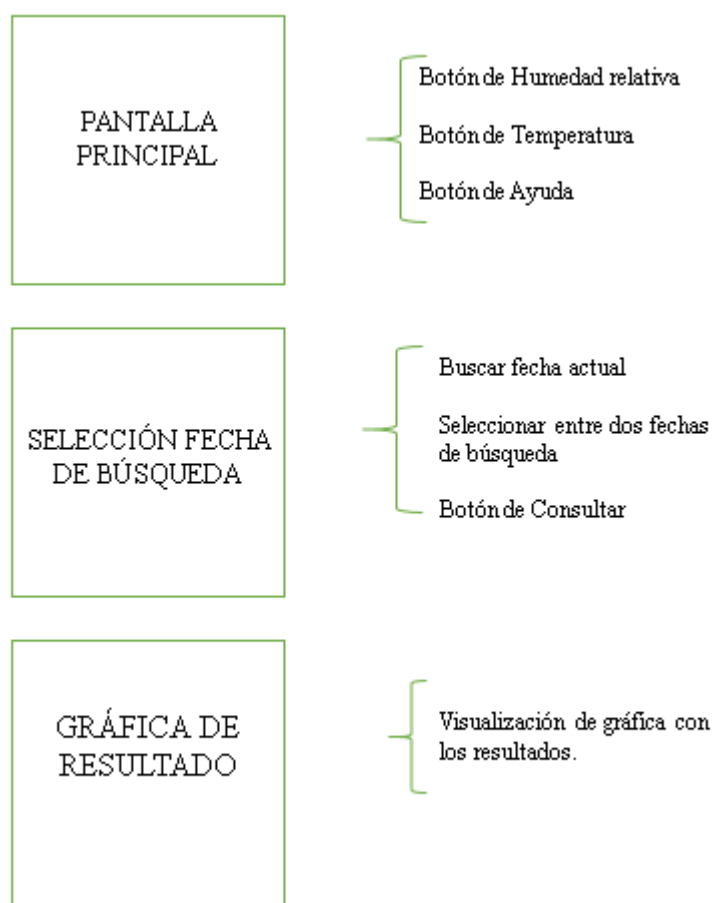


Figura 11. Diagrama funcional del desarrollo de software.

#### 4.2.1 Descripción del aplicativo móvil.

Agro App es un aplicativo móvil, el cual permite monitorear las variables de temperatura y humedad relativa de los cultivos bajo invernaderos o cámaras térmicas, las cuales se pueden visualizar en tiempo real o pueden ser consultadas por fecha de búsqueda, ya que la información se encuentra almacenada en una base de datos.

#### 4.2.2 Interfaz gráfica y descripción de los botones

La interfaz gráfica es amigable con el usuario, ya que es fácil de manejar. Cuenta con 3 pantallas: el menú principal, el panel de selección de fecha de consulta deseada y finalmente el resultado de búsqueda.

##### Menú principal

Como se puede apreciar en la siguiente imagen, el menú cuenta con 3 botones: Humedad Relativa, Temperatura y Ayuda.



Figura 12. Interfaz gráfica principal del aplicativo móvil

### Humedad Relativa



Humedad Relativa

Corresponde a la variable física de humedad relativa que desea conocerse del cultivo, si desea información acerca de esta variable se debe dar clic sobre el icono.

### Temperatura



Temperatura

Corresponde a la variable física temperatura que desea conocerse del cultivo, si desea información acerca de esta variable se debe dar clic sobre el icono.

Ayuda



### Ayuda

Es el botón que contiene las indicaciones para realizar la consulta en primera instancia, al presionarlo se visualiza un mensaje tipo toast que dice “Por favor seleccione la variable que desea consultar, debe dar click sobre el ícono que aparece en pantalla”

Luego de presionar cualquiera de las dos variables que se desean consultar, se pasa al siguiente menú para seleccionar la fecha de consulta.

### Seleccionar fecha

Para seleccionar la fecha de búsqueda se plantean dos opciones, una de ellas corresponde a ir a la fecha actual y la otra opción es seleccionar un rango de búsqueda, seleccionando dos fechas, una inicial y otra final. A continuación se puede observar la interfaz.



Figura 13. Segunda interfaz gráfica del aplicativo móvil.



#### **Ver hoy**

Permite consultar las variables de manera rápida, ubicando la fecha actual y accediendo directamente a los datos registrados.



#### **Fecha inicial**

Este botón deber ser presionado, cuando se va a consultar la información en un rango de fechas, este corresponde a la primera fecha, al seleccionarlo, se despliega el calendario que permite elegir la fecha deseada.



#### **Fecha final**

Este botón deber ser presionado, cuando se va a consultar la información en un rango de fechas, este corresponde a la segunda fecha, al seleccionarlo, se despliega el calendario que permite elegir la fecha deseada.

CONSULTAR

### Consultar

Una vez definido el rango de búsqueda con los dos botones anteriores, se debe seleccionar el botón consultar, para dar paso a la siguiente interfaz que es la encargada de mostrar los resultados.

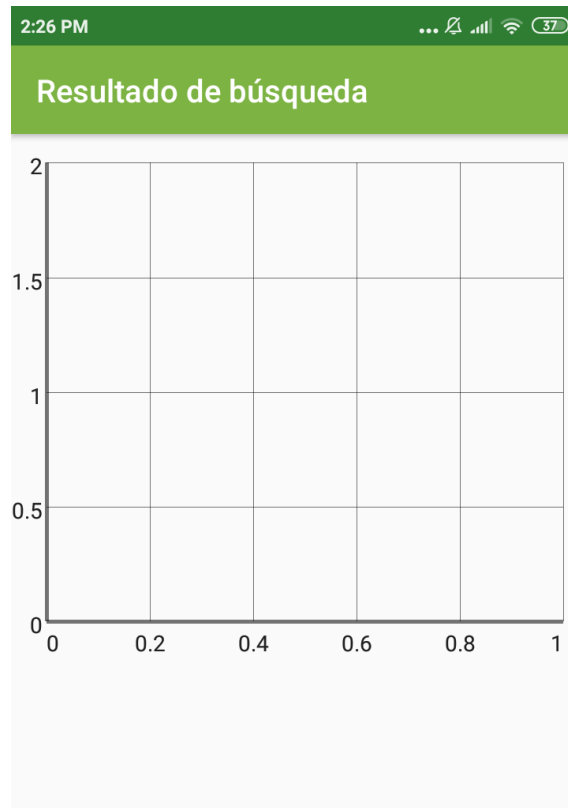


Figura 14. Resultado de búsqueda en el aplicativo móvil.

#### 4.2.3 Circuito implementado

En la siguiente Figura x, se puede observar el diagrama completo de la implementación final, en primer lugar se encuentra el sensor DHT22, el cual es el encargado de capturar las medidas de temperatura y humedad relativa del cultivo en el invernadero, y seguidamente son adquiridas por el sistema embebido BeagleBone Black (BBB) a través del pin de datos P9\_14. Se establece la conexión entre el módulo BeagleBone Black y el Servidor PC. Finalmente en el aplicativo móvil se pueden realizar las consultas de manera remota de las dos variables físicas del cultivo.

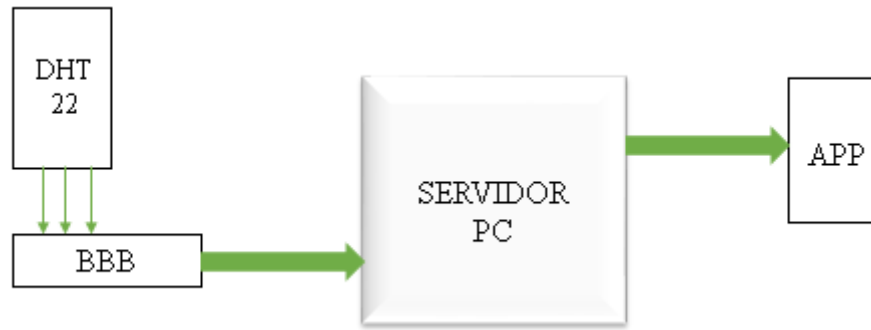


Figura 15. Diagrama del circuito implementado.



## 5. RESULTADOS



Figura 16. Implementación física en el invernadero Horfrubella.

Se realizaron pruebas en tres lugares con características diferentes: en la Universidad Tecnológica de Pereira, en el corregimiento Mundo Nuevo y finalmente en el invernadero Horfrubella ubicado en el corregimiento La bella en el municipio de Pereira, para verificar el correcto funcionamiento de la implementación, tanto del sistema de telemetría como el de la información, en la figura 16, se puede observar el montaje realizado en el corregimiento La bella en el municipio de Pereira.

En la siguiente figura 17, se observa la base de datos conformada por más de 400 registros tomados en diferentes horas del día, durante varios días. La toma de valores es de cada 5 minutos para ver la diferencia entre los datos. En la base de datos a continuación, se aprecia cómo va aumentando la humedad relativa y disminuyendo la temperatura, esto es debido a las precipitaciones que se presentaron en el momento de la toma de las variables físicas.

				id	temperatura	humedad	fecha
<input type="checkbox"/>		Editar		Copiar		Borrar	402 22.60 72.59 2019-03-18 15:45:10
<input type="checkbox"/>		Editar		Copiar		Borrar	403 21.10 78.59 2019-03-18 15:50:11
<input type="checkbox"/>		Editar		Copiar		Borrar	404 19.40 89.69 2019-03-18 15:55:11
<input type="checkbox"/>		Editar		Copiar		Borrar	405 18.79 95.80 2019-03-18 16:00:13
<input type="checkbox"/>		Editar		Copiar		Borrar	406 18.79 95.80 2019-03-18 16:05:13
<input type="checkbox"/>		Editar		Copiar		Borrar	407 18.70 98.80 2019-03-18 16:10:14
<input type="checkbox"/>		Editar		Copiar		Borrar	408 18.50 97.90 2019-03-18 16:15:15
<input type="checkbox"/>		Editar		Copiar		Borrar	409 18.79 97.30 2019-03-18 16:20:16
<input type="checkbox"/>		Editar		Copiar		Borrar	410 18.79 97.40 2019-03-18 16:25:17
<input type="checkbox"/>		Editar		Copiar		Borrar	411 18.79 98.40 2019-03-18 16:30:17
<input type="checkbox"/>		Editar		Copiar		Borrar	412 18.90 97.59 2019-03-18 16:35:18
<input type="checkbox"/>		Editar		Copiar		Borrar	413 18.79 97.59 2019-03-18 16:40:19
<input type="checkbox"/>		Editar		Copiar		Borrar	414 18.50 99.69 2019-03-18 16:45:19
	Consola	Editar		Copiar		Borrar	415 18.70 99.90 2019-03-18 16:50:20

Figura 17. Base de datos con los valores obtenidos del cultivo.

En la siguiente figura 18 se aprecia el resultado de búsqueda para la variable temperatura correspondiente a la fecha de 2019-03-18 hasta 2019-03-19

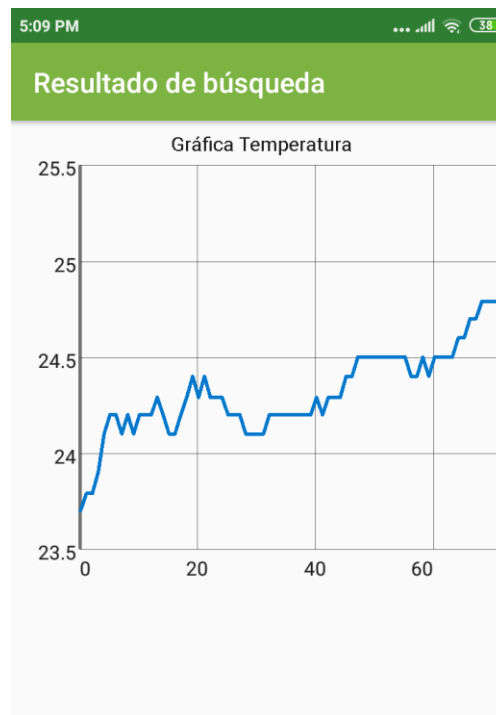


Figura 18. Resultado de búsqueda para la variable temperatura.

En la siguiente figura 19 se aprecia el resultado de búsqueda para la variable humedad relativa correspondiente a la fecha de 2019-03-18 hasta 2019-03-19.

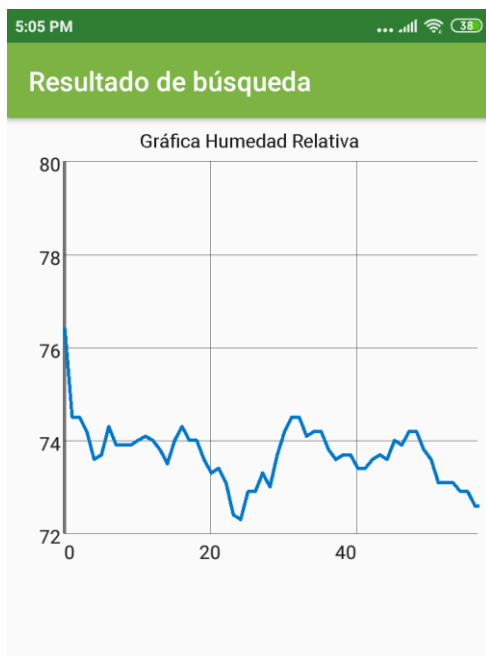


Figura 19. Resultado de búsqueda para la variable humedad relativa.

Cuando se realiza una consulta a través del aplicativo móvil y no hay conexión entre el dispositivo móvil y el servidor, aparece en pantalla un mensaje tipo toast, informando que “No hay conexión” como se aprecia en la siguiente figura 18.

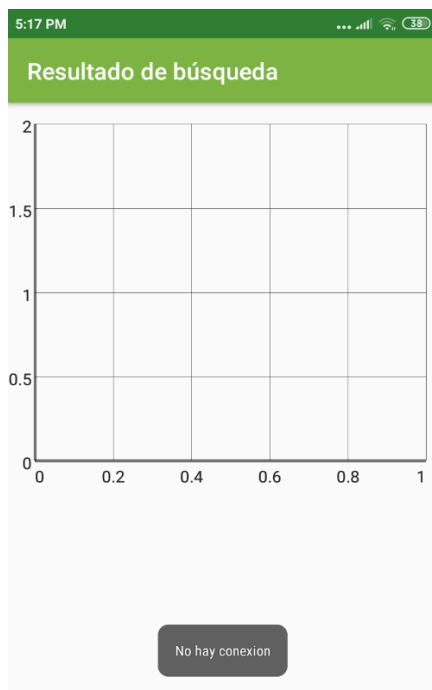


Figura 20. Resultado de búsqueda cuando no hay conexión.

Si en la fecha de búsqueda, no hay datos almacenados en el sistema de información, la aplicación no deja de funcionar, en la interfaz de resultados, informa a través de un mensaje que “No hay datos” como se puede ver en la siguiente figura 19.

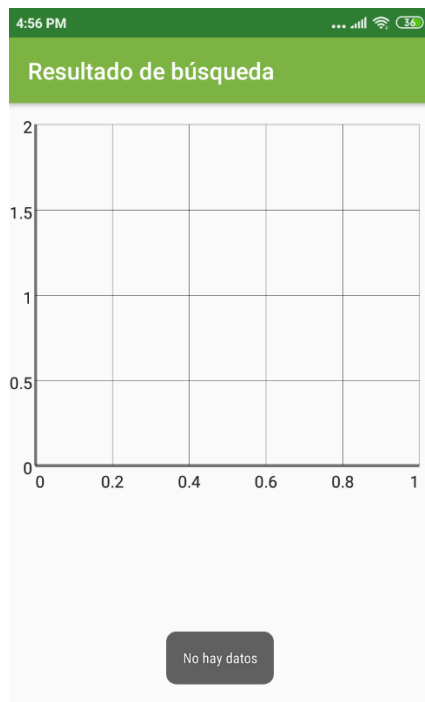


Figura 21. Resultado de búsqueda cuando no hay datos.

## **6. CONCLUSIONES**

- El sistema de monitoreo implementado es confiable, ya que en las pruebas realizadas, no demostró intermitencia, ni tiempos muertos y se garantizó el tiempo de captura.
- Al realizar las pruebas de usabilidad el aplicativo móvil demuestra robustez ya que no presenta errores ante las diferentes posibilidades que ingrese el usuario.
- Se implementó un sistema de adquisición y visualización de datos, el cual podría mejorar la producción de plantas en invernaderos ayudando a los agricultores colombianos.

## **7. TRABAJOS FUTUROS**

- Adicionar sensores que entreguen más información acerca del cultivo en cámaras térmicas, por ejemplo humedad del suelo y luminosidad.
- Implementar un sistema que permita no solo monitorear las variables de humedad relativa y temperatura, sino controlarlas para que permanezcan en un rango óptimo y de esta manera tener mayor productividad del cultivo.
- Implementar algoritmos para crear alertas tempranas ante posibles enfermedades.

## BIBLIOGRAFÍA

- [1] Tauger, Mark B, Agriculture in world history 2010 primer edición.
- [2] Frank tejada Cabrera, Origen y evolución de la agricultura 2013 [En línea] consultado el 14 de agosto de 2018 disponible en <http://hoy.com.do/origen-y-evolucion-de-la-agricultura/>
- [3] Recursos Naturales, [En Línea] consultado el 14 agosto de 2018 Disponible en <https://www.colombia.com/colombia-info/informacion-general/recursos-naturales/>
- [4] Colprensa EL universal “Los campesinos no estamos siendo reconocidos en el país” 31 mayo 2018. [En línea] Disponible en <http://www.eluniversal.com.co/colombia/los-campesinos-no-estamos-siendo-reconocidos-en-el-pais-279832>
- [5] Ortega Fernández, Luis Alfonso, et al. Implementación de Servicios Integrados Participativos de Clima para la Agricultura (PICSA) en el TESAC-Cauca Colombia. 2018.
- [6] I. F. García-Tejero, J. M. Costa, R. S. N. Lima, V. H. Durán-Zuazo, J. L. Muriel, M. M. Chaves, Aplicaciones de la termografía de infrarrojos en la agricultura moderna.
- [7] Adrián González, Gelberth Amarillo, Milton Amarillo, Francisco Sarmiento, Drones aplicados a la agricultura de precisión 2015.
- [8] Gobierno de Colombia MINTIC [En línea] Consultado el 6 de junio de 2018 Disponible en <http://www.mintic.gov.co>
- [9] José L. Fernández, Temperatura [En línea] consultado el 18 de marzo de 2019 Disponible en <https://www.fisicalab.com/apartado/temperatura#contenidos>
- [10]Canna research, [En línea] consultado el 18 de marzo de 2019. Disponible en [http://www.canna.es/influencia\\_temperatura\\_ambiental\\_en\\_las\\_plantas](http://www.canna.es/influencia_temperatura_ambiental_en_las_plantas)
- [11] Lourdes Huertas, El control ambiental en invernaderos.
- [12] Datasheet DHT22 [En línea] consultado el 6 de junio de 2018 Disponible en <https://www.mouser.com/ds/2/737/dht-932870.pdf>
- [13] BeagleBoard [En línea] Disponible en <https://beagleboard.org/black>
- [14] Developers [En línea] Disponible en <https://developer.android.com/studio/intro?hl=es-419>

- [15] Tutorial Kotlin Ya [En línea] Disponible en <http://tutorialesenpdf.com/kotlin/>
- [16] Charly 2012 [En línea] Consultado el 18 de marzo 2019 Disponible en <http://myu-charly.blogspot.com/>
- [17] agtech [En línea] consultado el 18 de marzo de 2018 Disponible en <http://agtech.cl/drones-para-mejorar-el-monitoreo-de-los-cultivos-de-invernadero/>
- [18] Hostname 2012, [En línea] consultado el 18 de marzo de 2018 Disponible en <https://www.hostname.cl/blog/que-es-phpmyadmin>
- [19] Andrés Castillo González, Diseño e implementación de un prototipo de una red de sensores inalámbricos para cultivos de frutas utilizando Beaglebone black. Disponible en repositorio UTP.
- [20] Jhonny Wbeimar Perea Palacios, Diseño de un sistema de monitoreo, registro y control de temperatura y humedad para un cultivo de invernadero. Disponible en repositorio UTP.



## ANEXOS

Anexo 1: MainActivity (Menú principal) en Android Studio lenguaje Kotlin

```
package com.example.utp.rodriquez.lina.agroappk
import android.content.Intent
import android.support.v7.app.AppCompatActivity
import android.os.Bundle
import android.widget.*
import kotlinx.android.synthetic.main.activity_main.*
```

```
class MainActivity : AppCompatActivity() {
    var tempe: ImageButton? = null

    var ayuda: Button? = null
    var hume: ImageButton? = null
    // var texto:TextView?=null

    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_main)
        tempe = findViewById(R.id.tempeButton)
        ayuda = findViewById(R.id.ayuda)
        hume = findViewById(R.id.humeButton)
        //hoy=findViewById(R.id.hoy)

        humeButton.setOnClickListener {

            val intent: Intent = Intent(this, SegundaActivity::class.java)
            decision=true
            startActivity(intent)
        }
    }
}
```

```

    }

    tempeButton.setOnClickListener {
        decision=false
        val intent: Intent = Intent(this, SegundaActivity::class.java)
        startActivity(intent)
    }

    // texto=findViewById(R.id.texto)

}

override fun onResume() {
    super.onResume()
    ayuda!!.setOnClickListener {
        Toast.makeText(
            applicationContext,
            "Por favor seleccione la variable que desea consultar, debe dar click sobre el
            ícono que aparece en pantalla",
            Toast.LENGTH_LONG
        ).show()
    }
}

companion object {
    var decision:Boolean=false
}
}

```

## Anexo 2. Segunda Activity en Android Studio en lenguaje Kotlin

```
package com.example.utp.rodriquez.lina.agroappk
```

```
import android.os.Bundle
import android.support.v7.app.AppCompatActivity
import android.app.DatePickerDialog
import android.content.Intent
import android.icu.text.SimpleDateFormat
import android.icu.util.Calendar
import android.view.View
import android.widget.Button
import android.widget.DatePicker
import android.widget.TextView
import android.widget.Toast
import com.android.volley.Request
import com.android.volley.Response
import com.android.volley.toolbox.StringRequest
import com.android.volley.toolbox.Volley
import kotlinx.android.synthetic.main.activity_main.*
import kotlinx.android.synthetic.main.activity_segunda.*
import kotlinx.android.synthetic.main.activity_tercera.*
import java.util.*
```

```
class SegundaActivity : AppCompatActivity() {
    var seleccion_fecha1: Button? = null
    var seleccion_fecha2: Button? = null
    var cal = Calendar.getInstance()
    var cal2 = Calendar.getInstance()
    var consulta: Button? = null
    var valida = false
    var valida1 = false
    var hoy: Button? = null

    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_segunda)
        consulta = findViewById(R.id.consultar)
        hoy = findViewById(R.id.hoy)

        hoy!!.setOnClickListener() {
            val currentTime = Calendar.getInstance().time
            val cal: Calendar
            cal = Calendar.getInstance()

```

```

        cal.add(Calendar.DATE,1)
        val myFormat = "yyyy-MM-dd" // mention the format you need
        val sdf = SimpleDateFormat(myFormat, Locale.US)
        ver_seleccion_fecha1!!.text=sdf.format(currentTime)
        ver_seleccion_fecha2!!.text=sdf.format(cal)

        val intent: Intent = Intent(this, TerceraActivity::class.java)
        startActivity(intent)

    }
    consultar.setOnClickListener {

        val intent: Intent = Intent(this, TerceraActivity::class.java)
        startActivity(intent)

    }

    ver_seleccion_fecha1 = findViewById(R.id.ver_seleccion_fecha_1)
    seleccion_fecha1 = findViewById(R.id.seleccion_fecha_1)

    ver_seleccion_fecha2 = this.ver_seleccion_fecha_2
    seleccion_fecha2 = this.seleccion_fecha_2

    ver_seleccion_fecha1!!.text = " De dd/mm/aaaa"
    ver_seleccion_fecha2!!.text = " Hasta dd/mm/aaaa"

    val dateSetListener1 = object : DatePickerDialog.OnDateSetListener {
        override fun onDateSet(
            view: DatePicker, year: Int, monthOfYear: Int,
            dayOfMonth: Int
        ) {
            cal2.set(Calendar.YEAR, year)
            cal2.set(Calendar.MONTH, monthOfYear)
            cal2.set(Calendar.DAY_OF_MONTH, dayOfMonth)
            valida1 = true
            updateDateInView()
        }
    }

    val dateSetListener = object : DatePickerDialog.OnDateSetListener {
        override fun onDateSet(
            view: DatePicker, year: Int, monthOfYear: Int,
            dayOfMonth: Int
        ) {
            cal.set(Calendar.YEAR, year)
            cal.set(Calendar.MONTH, monthOfYear)

```

```

        cal.set(Calendar.DAY_OF_MONTH, dayOfMonth)
        valida = true
        updateDateInView()
    }
}

seleccion_fecha1!!.setOnClickListener(object : View.OnClickListener {
    override fun onClick(view: View) {

        if (view == seleccion_fecha1) {
            DatePickerDialog(
                this@SegundaActivity,
                dateSetListener,
                cal.get(Calendar.YEAR),
                cal.get(Calendar.MONTH),
                cal.get(Calendar.DAY_OF_MONTH)

            ).show()
        }
    }
})

seleccion_fecha2!!.setOnClickListener(object : View.OnClickListener {
    override fun onClick(view: View) {

        if (view == seleccion_fecha2) {
            DatePickerDialog(
                this@SegundaActivity,
                dateSetListener1,
                cal2.get(Calendar.YEAR),
                cal2.get(Calendar.MONTH),
                cal2.get(Calendar.DAY_OF_MONTH)

            ).show()
        }
    }
})

}

private fun updateDateInView() {
    val myFormat = "yyyy-MM-dd" // mention the format you need
    val sdf = SimpleDateFormat(myFormat, Locale.US)
    ver_seleccion_fecha1!!.text = sdf.format(cal.getTime())
    ver_seleccion_fecha2!!.text = sdf.format(cal2.getTime())

    if (valida && valida1) {

```

```
        consultar.isEnabled = true
    }
}
companion object {
    var ver_seleccion_fecha1: TextView? = null
    var ver_seleccion_fecha2: TextView? = null
    var fechaHoy:String=""
}
}
```

### Anexo 3. Tercera Activity escrita en Kotlin

```
package com.example.utp.rodriquez.lina.agroappk

import android.graphics.Color
import android.os.Bundle
import android.support.v7.app.AppCompatActivity
import android.view.Display
import android.widget.Toast
import com.android.volley.Request
import com.android.volley.Response
import com.android.volley.toolbox.StringRequest
import com.android.volley.toolbox.Volley
import com.jjoe64.graphview.GraphView
import com.jjoe64.graphview.Viewport
import com.jjoe64.graphview.series.DataPoint
import com.jjoe64.graphview.series.LineGraphSeries
import com.jjoe64.graphview.series.Series
import kotlinx.android.synthetic.main.activity_tercera.*

import org.json.JSONObject

class TerceraActivity : AppCompatActivity() {

    var graf:GraphView?=null
    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_tercera)

        // Enables Always-on
        graf=findViewById(R.id.Grafica)
        graf!!.viewport.setScalableY(true)

    }
    override fun onResume() {
        super.onResume()
        val queue = Volley.newRequestQueue(this)
        val url =
            "http://192.168.69.39/cultivo/pru.php?fechaini=${SegundaActivity.ver_seleccion_fecha1!!}!!< i>text}&fechafin=${SegundaActivity.ver_seleccion_fecha2!!}!!< i>text}"
        var datos= LineGraphSeries<DataPoint>()

        //Request a string response from the provided URL.
```

```

val stringRequest = StringRequest(

    Request.Method.GET, url,
    Response.Listener<String> { response ->
        //Display the first 500 characters of the response string.
        try{
            val r=JSONObject(response)
            val arreglo=r.getJSONArray("datos")
            var arre= arrayOfNulls<DataPoint>(arreglo.length())
            for (i in 0 .. arreglo.length()-1 ) {
                val re = arreglo.getJSONObject(i)
                if (MainActivity.decision){
                    graf!!.title="Gráfica Humedad Relativa"
                    graf!!.title textSize=
                    val d= re.getString("humedad").toDouble()
                    arre[i]=(DataPoint(i.toDouble(),d))

                }
                else{
                    graf!!.title="Gráfica Temperatura"
                    val d= re.getString("temperatura").toDouble()
                    arre[i]=(DataPoint(i.toDouble(),d))

                }

            }

            //
            Toast.makeText(this,res.getString("temperatura"),Toast.LENGTH_LONG).show()
        }
        graf!!.addSeries(LineGraphSeries<DataPoint>(arre))

    }
    catch (e:Exception){
        Toast.makeText(this,"No hay datos",Toast.LENGTH_LONG).show()
    }
    },
    Response.ErrorListener { Toast.makeText(this, "No hay
conexion",Toast.LENGTH_LONG).show() })

    queue.add(stringRequest)

}
}

```



Anexo 4. Código en C++ para realizar la adquisición de datos y conectar con la base de datos.

```
#include "/home/debian/Adafruit_Python_DHT-  
master/source/Beaglebone_Black/bbb_dht_read.h"  
  
#include "/home/debian/Adafruit_Python_DHT-master/source/common_dht_read.h"  
  
#include "/home/debian/Adafruit_Python_DHT-  
master/source/Beaglebone_Black/bbb_mmio.h"  
  
#include <iostream>  
  
#include <unistd.h>  
  
#include <sys/stat.h>  
  
#include <fcntl.h>  
  
#include <cstdio>  
  
#include <errno.h>  
  
#include <cstdlib>  
  
#include <fstream>  
  
#include <cstring>  
  
#include <mysql/mysql.h>  
  
using namespace std;  
  
bool ConexionMysql(string CodMysqlQuery,string tabla);  
  
int main() {  
    float temp=0.0,hum=0.0;  
    string tabla="datos";  
    string h="humedad";  
    string t="temperatura";  
    string CodMysqlQuery;  
    while(1)  
    {  
        while((hum==0)&&(temp==0))  
        {
```

```

        int dato = bbb_dht_read(22,1,18,&hum,&temp);

        cout<<"Humedad "<<hum<<endl;

        cout<<"Temperatura "<<temp<<endl;

    }

    CodMysqlQuery="INSERT INTO "+tabla+"("+t+", "+h+")"+"
VALUES('"+to_string(temp)+"','"+to_string(hum)+"')";

    if (!ConexionMysql(CodMysqlQuery,tabla)){

        cout << "ERROR de envio" << endl;

    }

    temp=0.0;
    hum=0.0;
    CodMysqlQuery.clear();
    sleep(300);
}

return 0;

}

bool ConexionMysql(string CodMysqlQuery,string tabla)
{
    const char* MY_HOSTNAME="192.168.0.113";
    const char* MY_DATABASE="cultivo";
    const char* MY_USERNAME="lina";
    const char* MY_PASSWORD="linautp";
    const char* MY_SOCKET=NULL;
    char* cstr=new char[CodMysqlQuery.length() + 1];
    enum {
        MY_PORT_NO = 3306,
        MY_OPT     = 0
    };

    MYSQL *conn;

```

```

conn = mysql_init(NULL);

if (!mysql_real_connect(
    conn,
    MY_HOSTNAME, MY_USERNAME,
    MY_PASSWORD, MY_DATABASE,
    MY_PORT_NO, MY_SOCKET, MY_OPT))
{
    cerr << mysql_error(conn) << endl;
    return false;
}

//cout<<CodMysqlQuery<<endl<<endl;

strcpy(cstr, CodMysqlQuery.c_str());
size_t st_len1 = strlen(cstr);
if(mysql_real_query(conn,cstr,st_len1))
{
    cout<<"Error Primer envio"<<endl;
    mysql_close(conn);
    return false;
}
else{ mysql_close(conn);
return true;
}
}

```

## Anexo 5 Establecer conexión con la base de datos

```
<?PHP

$hostname = "localhost";
$database = "cultivo";
$username = "root";
$password = "";
$d=0;
$info = array();

    if (isset($_GET["fechaini"])&& isset($_GET["fechafin"])){
        $fechaini=$_GET["fechaini"];
        $fechafin=$_GET["fechafin"];
        $conexion = mysqli_connect($hostname,$username,$password,$database);

        $consulta = "SELECT * FROM datos WHERE fecha BETWEEN
'{$fechaini}' AND '{$fechafin}' ";
        $resultado = mysqli_query($conexion,$consulta);
        if($resultado){
            while($reg=mysqli_fetch_assoc($resultado)){
                $info['datos'][]=($reg);

            }
            mysqli_close($conexion);
            echo json_encode($info);

        }

    }
```